**Introduction**

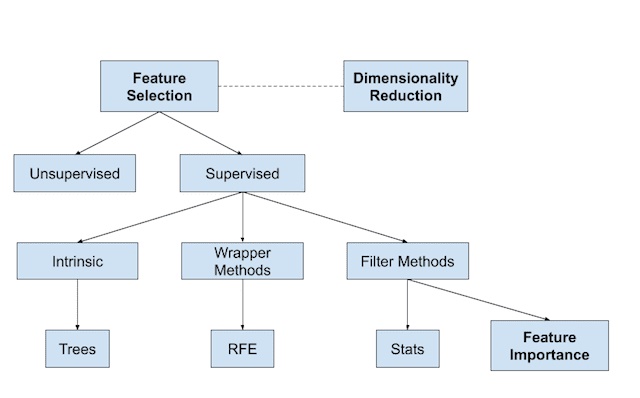
In the proposed project, the main objective is to classify the memorable and not memorable pics of the image dataset. For the classification of targeted images, we used the tree-based machine learning model. As in the tree-based learning approach, Random Forest model is fast and feasible for high dimensional data [1], [2], we trained the random forest model for the classification of memorable and not memorable images.

**Methodology**

**Dataset:** For the proposed study, image features dataset was used that was consist of 4608 features. The features were extracted from the deep learning models. CNN extracted the 4096 features while the rest of the 512 features were the gist features. The selected dataset has the class column that have the two unique label 0 and 1 for not memorable and memorable images respectively.

**Preprocessing:** As the dataset was based on the 3 different files: training1.csv, training2.csv and test.csv. Firstly, the training1.csv and training2.csv have the same features set were merged into single training set. Secondly, the null and NAN value in the training and testing set were replaced by the mean value of that image. For this, the mean of each row was calculated and then replace by the null or NAN value of that row/image-feature. After removing the null values, the training and testing dataset was in clean and normalized form and didn’t require further scaling.

**Feature Selection:** As the selected dataset is a high dimensional dataset that require the dimensionality reduction for the proper learning of the machine learning model. Firstly, we remove the confidence label feature as it was calculated manually base on the human prediction and it was not feasible for the runtime prediction of the images. For the rest of features selection, we used the feature importance technique from the three available techniques available for classification. The overview of feature selection techniques is also show in the Figure # (fetch from [3]). For the calculation of feature importance, random forest model was used that calculate the importance score of each feature. Later, all the features were sorted in ascending order and then extract the top 50, 100, 200, 500 and 1000 important features for the classification of memorable and not memorable images.



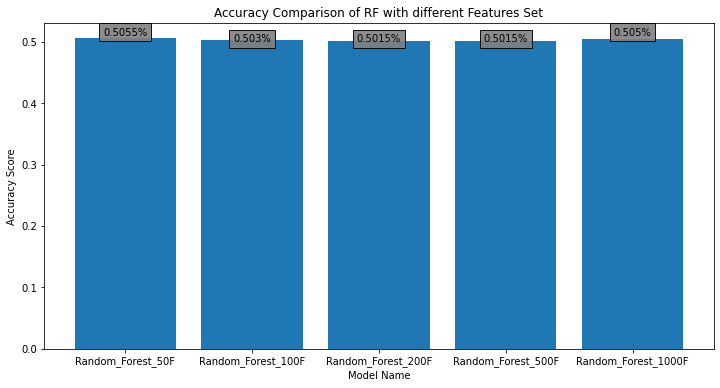
**Train Test Split:**  The dataset was already base on the two files: training.csv and testing.csv that contain the 3400 and 2000 samples respectively. We used the training set for the training of the model while test set for the testing/evaluation of the model.

**Training:** For the prediction of the memorable and not memorable images, random forest model was used for the training on the image’s features. Random Forest model is feasible for high dimensional data. It is also a fast model because it working on the subset of the features. Random Forest model is also able to handle the outliers and find the complex relationship between the variables [1], [2]. For the training of the model, training features and training labels were passed to the model with default parameters. Further, different hyper parameters including n-estimator, max-depth, and bootstrap value were tuned to achieve the maximum performance of the model.

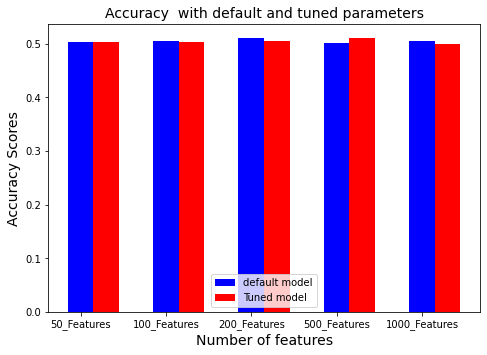
**Evaluation Measures:** The well-known evaluation measure “Accuracy” was used for the evaluation of the trained model. Accuracy of the model was calculated by using the test set.

**Results and Discussion**

The selected model “Random Forest” was trained with the default parameter values. Different number of training set with 50, 100, 200, 500 and 1000 imported features were used to train the model. Test set was used to evaluate the models using accuracy evaluation measure. The accuracy of different training sets with default parameters is show in figure #. The figure # showed the model’s name with different set based on 50 features to 1000 features along x-axis while accuracy score of test set along y-axis.



Next, the random forest was tuned by using hyper parameter tuning technique to increase the performance of the model. The n-estimator, bootstrap and max-depth parameter was tuned for different values. The result of same training sets with 50 to 1000 features set, showed the slightly increased accuracy. The performance comparison of models on test set is also shown in figure #. Figure # showed the difference in accuracy with default and tuned parameters on the training of same training sets.



As the accuracy is good evaluation measure for balance dataset, but the accuracy is not good evaluation measure for class imbalance dataset. So, we also calculated the precision, recall and F1-score on testing set for every train model. The complete evaluation report of trained models is also presented in Table 1. The Table 1 showed the model’s name in the first column followed by their accuracy, precision, recall and F1-Score.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | accuracy | precision | recall | f1-score |
| Random Forest 50F | 0.5030 | 0.503072 | 0.503074 | 0.502960 |
| Random Forest 100F | 0.5045 | 0.504581 | 0.504584 | 0.504445 |
| Random Forest 200F | 0.5110 | 0.511096 | 0.511107 | 0.510917 |
| Random Forest 500F | 0.5010 | 0.501138 | 0.501140 | 0.500800 |
| Random Forest 1000F | 0.5050 | 0.505108 | 0.505115 | 0.504889 |
| Tuned Random Forest 50F | 0.5035 | 0.503503 | 0.503503 | 0.503497 |
| Tuned Random Forest 100F | 0.5030 | 0.503090 | 0.503093 | 0.502928 |
| Tuned Random Forest 200F | 0.5060 | 0.506138 | 0.506151 | 0.505802 |
| Tuned Random Forest 500F | 0.5115 | 0.511623 | 0.511643 | 0.511350 |
| Tuned Random Forest 1000F | 0.4990 | 0.499144 | 0.499142 | 0.498779 |

The results showed that the random forest outer perform on training set of 500 features with tuned parameters. But the accuracy score of the best model (0.5115%) is not satisfactory. It is due to the different factors like the high dimensionality, too much NAN and zero values. We try to rescue these limitations by feature selection and replacement of NAN by the mean value of the image. But the majority of features value contain the zero. It seems that high dropout rate was used during the extraction of feature from CNN that replace majority value by zero and lead to the loss of important information in an image. Ultimately, the models are unable to established a relation between input variable and label. By using all the preprocessing and evaluation technique, we hypothesize that the dataset have the lack of information for ML models due to the majority of zero value and need to extract the features again with low dropout rate to gain suitable results.